**Relatório de Justificativa  
Padrão SAGA**

|  |  |
| --- | --- |
| **Empresa:** | **SOAT** |
| **Versão:** | 1.0 |
| **Data da última modificação:** | 03/07/2025 |
| **Autor:** | Vernon Araújo Corrêa Simões |

|  |
| --- |
| **Descritivo da aplicação:** |
| Criar a API, para que posteriormente o time de frontend integre a solução. O desenho da solução envolve as seguintes necessidades do negócio: • Cadastrar um veículo para venda (Marca, modelo, ano, cor, preço); • Editar os dados do veículo; • Efetuar a venda de um veículo (somente para compradores cadastrados); • Listagem de veículos à venda, ordenada por preço, do mais barato para o mais caro; • Listagem de veículos vendidos, ordenada por preço, do mais barato para o mais caro; • Cadastro de compradores. |

1. **Objetivo**

Garantir **consistência eventual e tolerância a falhas** nas operações que envolvem múltiplas etapas ou serviços na plataforma de revenda de veículos — especialmente no processo de **venda de um veículo**, onde é necessário:

* Verificar se o veículo está disponível;
* Registrar a venda;
* Atualizar o status do veículo;
* Associar o CPF do comprador;
* Persistir as informações com segurança.

|  |  |  |
| --- | --- | --- |
| **Tipo de SAGA** | **Característica** | **Exemplo** |
| Orquestrada | Um **coordenador central** comanda as etapas | Orquestrador chama cada serviço na ordem correta |
| Coreografada | Cada serviço **reage a eventos** e executa ações | Venda iniciada → Atualiza estoque → Notifica cliente |

1. **Justificativa Técnica: Orquestrada**

**Simplicidade operacional:**

* Como a aplicação utilizará o AWS Lambda e arquitetura hexagonal com serviços bem isolados, um **coordenador central** (ex: um Lambda orquestrador ou um gerenciador de caso de uso) **pode controlar o fluxo completo da venda** com clareza.

**Visibilidade e controle:**

* É mais fácil **logar, monitorar e auditar** os passos de uma saga quando há um componente orquestrador explícito.

**Gerenciamento de falhas centralizado:**

* Caso a persistência falhe ou o status do veículo não seja alterado com sucesso, é possível aplicar uma **compensação coordenada**, como desfazer a venda.

**Volume e complexidade baixos:**

* O processo de venda envolve poucos passos e serviços (atualmente todos centralizados na API), então não há necessidade de desacoplamento complexo via eventos.

**Fluxo da Venda de Veículo (SAGA):**

1. **Cliente escolhe o veículo**
2. **Orquestrador** pede para o veiculos-service **reservar**
3. Se OK, o orquestrador pede ao pagamentos-service **gerar código**
4. Cliente **paga** → pagamentos-service confirma
5. Orquestrador manda veiculos-service dar baixa no estoque
6. Venda finalizada

Se algo der errado (ex: pagamento falha), o orquestrador **cancela a reserva**.